Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Выборнова Любовь Алексеевна
Должность: Ректор
Дата подписания: 30.10.2020 09:57:59
Уникальный программный ключ:
c3b3b9c625f6c113afa2a2c42baff9e05a38b76e

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

«Поволжский государственный университет сервиса» (ФГБОУ ВО «ПВГУС»)

Кафедра «Информационный и электронный сервис»

РАБОЧАЯ ПРОГРАММА МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК.05.02 «РАЗРАБОТКА КОДА ИНФОРМАЦИОННЫХ СИСТЕМ»

Специальность **09.02.07 «Информационные системы и программирование»**

Тольятти 2020

2

Рабочая программа междисциплинарного курса «*Разработка кода информационных систем*» разработана в соответствии с Федеральным государственным образовательным стандартом среднего профессионального образования по специальности *09.02.07 «Информационные системы и программирование»*, утвержденным приказом Министерства образования и науки Российской Федерации от 9 декабря 2016 года № 1547.

Разработчик РПД:

к.т.н., доцент		Т.С. Яницкая
(ученая степень, ученое звание)	(подпись)	(ФИО)

СОГЛАСОВАНО:

Директор научной библиотеки		В.Н. Еремина
	(подпись)	(ФИО)
Начальник управления по информатизации		В.В. Обухов
	(подпись)	(ФИО)

РПД утверждена на заседании кафедры «Информационный и электронный сервис»

« 27 » декабря 20 19 г., протокол № 5

Заведующий кафедрой, д.т.н., профессор		В.И. Воловач
(уч.степень, уч.звание)	(подпись)	(ФИО)

СОГЛАСОВАНО:

Начальник учебно-методического отдела		Н.М. Шемендюк
	(подпись)	(ФИО)

Рабочая программа дисциплины утверждена в составе основной профессиональной образовательной программы решением Учёного совета. Протокол №4 от 22.01.2020г.

Рабочая программа дисциплины актуализирована и утверждена в составе образовательной программы решением Ученого совета от 23.09.2020 г. Протокол №3

1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

1.1. Цель освоения междисциплинарного курса

Целью освоения междисциплинарного курса является формирование у обучающихся следующих компетенций:

Код компетенции	Наименование компетенции
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.
ОК 02	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
ОК 03	Планировать и реализовывать собственное профессиональное и личностное развитие.
ОК 04	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
ОК 09	Использовать информационные технологии в профессиональной деятельности.
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языке.
ПК 5.1	Собирать исходные данные для разработки проектной документации на информационную систему.
ПК 5.2	Разрабатывать проектную документацию на разработку информационной системы в соответствии с требованиями заказчика.
ПК 5.3	Разрабатывать подсистемы безопасности информационной системы в соответствии с техническим заданием.
ПК 5.4	Производить разработку модулей информационной системы в соответствии с техническим заданием.
ПК 5.5	Осуществлять тестирование информационной системы на этапе опытной эксплуатации с фиксацией выявленных ошибок кодирования в разрабатываемых модулях информационной системы.
ПК 5.6	Разрабатывать техническую документацию на эксплуатацию информационной системы.
ПК 5.7	Производить оценку информационной системы для выявления возможности ее модернизации.

1.2. Планируемые результаты освоения дисциплины

В результате освоения дисциплины обучающийся должен:

иметь практический опыт в:

управлении процессом разработки приложений с использованием инструментальных средств;

обеспечении сбора данных для анализа использования и функционирования информационной системы;

программировании в соответствии с требованиями технического задания;

использовании критериев оценки качества и надежности функционирования информационной системы;

применении методики тестирования разрабатываемых приложений;

определении состава оборудования и программных средств разработки информационной системы;

разработке документации по эксплуатации информационной системы;

проведении оценки качества и экономической эффективности информационной системы в рамках своей компетенции;

модификации отдельных модулей информационной системы.

уметь:

осуществлять постановку задач по обработке информации;

проводить анализ предметной области;

осуществлять выбор модели и средства построения информационной системы и программных средств;

использовать алгоритмы обработки информации для различных приложений;

решать прикладные вопросы программирования и языка сценариев для создания программ;

разрабатывать графический интерфейс приложения;

создавать и управлять проектом по разработке приложения;

проектировать и разрабатывать систему по заданным требованиям и спецификациям.

знать:

основные виды и процедуры обработки информации, модели и методы решения задач обработки информации;

основные платформы для создания, исполнения и управления информационной системой;

основные процессы управления проектом разработки;

основные модели построения информационных систем, их структуру, особенности и области применения;

методы и средства проектирования, разработки и тестирования информационных систем;

систему стандартизации, сертификации и систему обеспечения качества продукции.

1.3. Место междисциплинарного курса в структуре образовательной программы

Междисциплинарный курс «Разработка кода информационных систем» относится к профессиональному циклу основной профессиональной образовательной программы.

2. СТРУКТУРА И СОДЕРЖАНИЕ МЕЖДИСЦИПЛИНАРНОГО КУРСА

2.1. Объём учебного междисциплинарного курса и виды учебной работы

Общая трудоёмкость междисциплинарного курса составляет **174 часа.** Их распределение по видам работ представлено в таблице:

Виды учебных занятий и работы обучающихся	**Трудоемкость, час**		
	всего	**7 семестр**	**8 семестр**
Общая трудоемкость	**174**	**60**	**114**
Объем работы обучающихся во взаимодействии с преподавателем по видам учебных занятий (всего), в т.ч.:	**28**	**14**	**14**
лекции	**12**	**6**	6
лабораторные работы	**-**	**-**	-
практические занятия	**12**	**6**	6
курсовое проектирование (консультации)	**-**	**-**	-
Самостоятельная работа	**146**	**46**	**100**
Контроль (часы на экзамен, зачет, контрольную работу)	**4**	**2**	**2**
Консультация перед экзаменом	**-**	**-**	**-**
Промежуточная аттестация		**Контрольная работа**	**Дифференцированный зачет**

2.2. Содержание междисциплинарного курса, структурированное по темам

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы			Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем		Самостоятельная работа, час	
		Лекции, час	Практические работы, час		
7 семестр					
ОК 01, ОК 02, ОК 03, ОК 04, ОК 09, ОК 10, ПК 5.1, ПК 5.2, ПК 5.3, ПК 5.4, ПК 5.5, ПК 5.6, ПК 5.7	**Тема 1. Основные инструменты для создания, исполнения и управления информационной системой.** Содержание темы: 1. Структура CASE-средства. Структура среды разработки. Основные возможности. 2. Основные инструменты среды для создания, исполнения и управления информационной системой. Выбор средств обработки информации. 3. Организация работы в команде разработчиков. Система контроля версий: совместимость, установка, настройка. 4. Обеспечение кроссплатформенности информационной системы. 5. Сервисно - ориентированные архитектуры. 6. Интегрированные среды разработки для создания независимых программ. 7. Особенности объектно-ориентированных и структурных языков программирования. 8. Разработка сценариев с помощью специализированных языков.	6			Доклад/сообщение, тестирование по темам лекционных занятий, отчет по практическим работам
	Практическая работа №1. Построение диаграммы Вариантов использования и диаграммы. Последовательности и генерация кода. Практическая работа №2. Построение диаграммы Кооперации и диаграммы Развертывания и генерация кода. Практическая работа №3. Построение диаграммы Деятельности, диаграммы Состояний и диаграммы Классов и генерация кода. Практическая работа №4. Построение диаграммы компонентов и генерация кода. Практическая работа №5. Построение диаграмм потоков данных и генерация кода.		6		
	Самостоятельная работа обучающихся, в т.ч.: 1. Работа с литературой, подготовка к занятиям, доработка практических работ.			46	
	ИТОГО за 7 семестр	**6**	**6**	**46**	

6 семестр					
ОК 01, ОК 02, ОК 03, ОК 04, ОК 09, ОК 10, ПК 5.1, ПК 5.2, ПК 5.3, ПК 5.4, ПК 5.5, ПК 5.6, ПК 5.7	**Тема 2. Разработка и модификация информационных систем.** Содержание темы: 1. Обоснование и осуществление выбора модели построения или модификации информационной системы. 2. Обоснование и осуществление выбора средства построения информационной системы и программных средств. 3. Построение архитектуры проекта. Шаблон проекта. 4. Определение конфигурации информационной системы. Выбор технических средств. 5. Формирование репозитория проекта, определение уровня доступа в системе контроля версий. Распределение ролей. 6. Настройки среды разработки. 7. Мониторинг разработки проекта. Сохранение версий проекта. 8. Требования к интерфейсу пользователя. Принципы создания графического пользовательского интерфейса (GUI). 9. Понятие спецификации языка программирования. Синтаксис языка программирования. Стиль программирования. 10. Основные конструкции выбранного языка программирования. Описание переменных, организация ввода-вывода, реализация типовых алгоритмов. 11. Создание сетевого сервера и сетевого клиента. 12. Разработка графического интерфейса пользователя. 13. Отладка приложений. Организация обработки исключений. 14. Виды, цели и уровни интеграции программных модулей. 15. Выбор источников и приемников данных, сопоставление объектов данных. 16. Транспортные протоколы. Стандарты форматирования сообщений. 17. Организация файлового ввода-вывода. 18. Процесс отладки. Отладочные классы. 19. Спецификация настроек типовой ИС.	6			Доклад/сообщение, тестирование по темам лекционных занятий, отчет по практическим работам
	Практическая работа №1. Обоснование выбора технических средств. Практическая работа №2. Стоимостная оценка проекта. Практическая работа №3. Построение и обоснование модели проекта. Практическая работа №4. Установка и настройка системы контроля версий с разграничением ролей. Практическая работа №5. Проектирование и разработка интерфейса пользователя. Практическая работа №6. Разработка графического интерфейса пользователя. Практическая работа №7. Реализация алгоритмов обработки числовых данных. Отладка приложения. Практическая работа №8. Реализация алгоритмов поиска. Отладка Приложения. Практическая работа №9. Реализация обработки табличных данных. Отладка приложения. Практическая работа №10. Разработка и отладка генератора случайных символов. Практическая работа №11. Разработка приложений для моделирования процессов и явлений.		6		

	Отладка приложения. Практическая работа №12. Интеграция модуля в информационную систему. Практическая работа №13. Программирование обмена сообщениями между модулями. Практическая работа №14. Организация файлового ввода-вывода данных. Практическая работа №15. Разработка модулей экспертной системы. Практическая работа №16. Создание сетевого сервера и сетевого клиента.				
	Самостоятельная работа обучающихся, в т.ч.: 1. Работа с литературой, подготовка к занятиям, доработка практических работ.			100	
	ИТОГО за 6 семестр	**6**	**6**	**100**	

2.3. Формы и критерии текущего контроля успеваемости (технологическая карта для студентов очной формы обучения)

Формы текущего контроля	Количество контрольных точек	Количество баллов за 1 контр. точку	Макс. возм. кол-во баллов
Доклад/сообщение	2	15	30
Тестирование по темам лекционных занятий	2	20	40
Отчет по практическим работам	1	30	30
		Итого по дисциплине	**100 баллов**

2.4. Шкала оценки результатов освоения дисциплины, сформированности результатов обучения

Форма проведения промежуточной аттестации	Условия допуска	Шкалы оценки уровня сформированности результатов обучения		Шкала оценки уровня освоения дисциплины		
		Уровневая шкала оценки компетенций	100 бальная шкала, %	100 бальная шкала, %	5-балльная шкала, дифференцированная оценка/балл	недифференцированная оценка
Контрольная работа, дифференцированный зачёт (по результатам накопительного рейтинга или в форме компьютерного тестирования).	допускаются все студенты	допороговый	ниже 61	ниже 61	«неудовлетворительно» / 2	не зачтено
		пороговый	61-85,9	61-69,9	«удовлетворительно» / 3	зачтено
				70-85,9	«хорошо» / 4	зачтено
		повышенный	86-100	86-100	«отлично» / 5	зачтено

3. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

3.1. Общие методические рекомендации по освоению дисциплины, образовательные технологии

Дисциплина реализуется посредством проведения контактной работы с обучающимися (включая проведение текущего контроля успеваемости), самостоятельной работы обучающихся и промежуточной аттестации.

Контактная работа может быть аудиторной, внеаудиторной, а также проводиться в электронной информационно-образовательной среде университета (далее - ЭИОС). В случае проведения части контактной работы по дисциплине в ЭИОС (в соответствии с расписанием учебных занятий), трудоемкость контактной работа в ЭИОС эквивалентна аудиторной работе.

При проведении учебных занятий по междисциплинарному курсу обеспечивается развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (включая проведение интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализ ситуаций и имитационных моделей, преподавание дисциплины в форме курса, составленного на основе результатов научных исследований, проводимых университетом, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей).

Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

- *балльно-рейтинговая технология оценивания;*
- *электронное обучение*.

Для оценки знаний, умений, навыков и уровня сформированности компетенции по дисциплине применяется балльно-рейтинговая система контроля и оценки успеваемости студентов. В основу балльно-рейтинговой системы положены принципы, в соответствии с которыми формирование рейтинга студента осуществляется в ходе текущего контроля успеваемости. Максимальное количество баллов в семестре – 100.

По итогам текущей успеваемости студенту может быть выставлена оценка по промежуточной аттестации в соответствии за набранными за семестр баллами. Студентам, набравшим в ходе текущего контроля успеваемости по дисциплине от 61 до 100 баллов и выполнившим все обязательные виды запланированных учебных занятий, по решению преподавателя без прохождения промежуточной аттестации выставляется оценка в соответствии со шкалой оценки результатов освоения дисциплины.

Результат обучения считается сформированным (повышенный уровень), если теоретическое содержание курса освоено полностью; при устных собеседованиях студент исчерпывающе, последовательно, четко и логически стройно излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, требующих применения знаний, использует в ответе дополнительный материал; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 86 до 100, что соответствует повышенному уровню сформированности результатов обучения.

Результат обучения считается сформированным (пороговый уровень), если теоретическое содержание курса освоено полностью; при устных собеседованиях студент последовательно, четко и логически стройно излагает учебный материал; справляется с задачами, вопросами и другими видами заданий, требующих применения знаний; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 61 до 85,9, что соответствует пороговому уровню сформированности результатов обучения.

Результат обучения считается несформированным, если студент при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет задания, не демонстрирует необходимых умений, качество выполненных заданий не соответствует установленным требованиям, качество их выполнения оценено числом баллов ниже 61, что соответствует допороговому уровню.

3.2. Методические указания по самостоятельной работе обучающихся

Самостоятельная работа обеспечивает подготовку обучающегося к аудиторным занятиям и мероприятиям текущего контроля и промежуточной аттестации по изучаемой дисциплине. Результаты этой подготовки проявляются в активности обучающегося на занятиях и в качестве выполненных практических заданий и других форм текущего контроля.

При выполнении заданий для самостоятельной работы рекомендуется проработка материалов лекций по каждой пройденной теме, а также изучение рекомендуемой литературы, представленной в Разделе 4.

В процессе самостоятельной работы при изучении дисциплины студенты могут использовать в специализированных аудиториях для самостоятельной работы компьютеры, обеспечивающему доступ к программному обеспечению, необходимому для изучения дисциплины, а также доступ через информационно-телекоммуникационную сеть «Интернет» к электронной информационно-образовательной среде университета (ЭИОС) и электронной библиотечной системе (ЭБС), где в электронном виде располагаются учебные и учебно-методические материалы, которые могут быть использованы для самостоятельной работы при изучении дисциплины.

Для обучающихся по заочной форме обучения самостоятельная работа является основным видом учебной деятельности.

4. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ МЕЖДИСЦИПЛИНАРНОГО КУРСА

4.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения междисциплинарного курса

Основная литература:

1. Душин, В. К. Теоретические основы информационных процессов и систем [Электронный ресурс] : учеб. для вузов по направлению "Информ. системы" и по специальностям "Информ. системы и технологии", "Сервис БРЭА", "Информ. сервис", "Сервис компьютерной и микропроцессорной техники", "Сервис" / В. К. Душин. - 5-е изд. - Документ Bookread2. - М. : Дашков и К, 2018. - 348 с. - Режим доступа: http://znanium.com/bookread2.php?book=450784.

2. Федорова, Г. Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности [Электронный ресурс] : учеб. пособие (09.02.05 Приклад. информатика (по отраслям) для проф. образоват. орг. / Г. Н. Федорова. - Документ Bookread2. - М. : Курс [и др.], 2017. - Режим доступа: http://znanium.com/bookread2.php?book=791799.

Дополнительная литература:

3. Золотухина, Е. Б. Управление жизненным циклом информационных систем. Продвинутый курс [Электронный ресурс] : крат. конспект лекций / Е. Б. Золотухина, С. А. Красникова, А. С. Вишня. - Документ Bookread2. - М. : Курс [и др.], 2017. - 119 с. - Режим доступа: http://znanium.com/bookread2.php?book=767219.

4.2. Профессиональные базы данных, информационно-справочные системы, интернет-ресурсы

1. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс]. - Режим доступа: http://elibrary.ru/defaultx.asp. - Загл с экрана.
2. Открытое образование [Электронный ресурс]. - Режим доступа: https://openedu.ru/. - Загл с экрана.
3. Университетская информационная система Россия [Электронный ресурс]. - Режим доступа: http://uisrussia.msu.ru/. – Загл. с экрана.
4. Электронная библиотечная система Поволжского государственного университета сервиса [Электронный ресурс]. - Режим доступа: http://elib.tolgas.ru/. - Загл. с экрана.
5. Электронно-библиотечная система Znanium.com [Электронный ресурс]. - Режим доступа: http://znanium.com/. – Загл. с экрана.
6. Электронно-библиотечная система Лань [Электронный ресурс]. - Режим доступа: https://e.lanbook.com/books. - Загл. с экрана.

4.3. Программное обеспечение

Информационное обеспечение учебного процесса по дисциплине осуществляется с использованием следующего программного обеспечения (лицензионного и свободно распространяемого), в том числе отечественного производства:

№ п/п	Наименование	Условия доступа
1	Microsoft Windows	из внутренней сети университета (лицензионный договор)
2	Microsoft Office	из внутренней сети университета (лицензионный договор)
3	All Fusion Process Modeler (BPwin)	из внутренней сети университета (лицензионный договор)
4	ArgoUML (StarUML)	из любой точки, в которой имеется доступ к сети Интернет (свободно распространяемое)
5	СДО MOODLE	из любой точки, в которой имеется доступ к сети Интернет (лицензионный договор)
6	JDK	из любой точки, в которой имеется доступ к сети Интернет (свободно распространяемое)
7	NetBeans (Eclipse)	из любой точки, в которой имеется доступ к сети Интернет (свободно распространяемое)

5. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО МЕЖДИСЦИПЛИНАРНОМУ КУРСУ

Специальные помещения представляют собой учебные аудитории для проведения занятий всех видов, предусмотренных образовательной программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы, мастерские и лаборатории, оснащенные оборудованием, техническими средствами обучения и материалами, учитывающими требования международных стандартов.

Занятия лекционного типа. Учебные аудитории для занятий лекционного типа укомплектованы мебелью и техническими средствами обучения, служащими для представления учебной информации (стационарные или переносные наборы демонстрационного оборудования (проектор, экран, компьютер/ноутбук), учебно-наглядные пособия (презентации по темам лекций), обеспечивающие тематические иллюстрации, соответствующие данной программе дисциплины.

Занятия семинарского типа. Учебные аудитории для занятий семинарского типа укомплектованы мебелью и техническими средствами обучения, служащими для представления учебной информации (стационарные или переносные наборы демонстрационного оборудования (проектор, экран, компьютер/ноутбук).

Промежуточная аттестация. Для проведения промежуточной аттестации по дисциплине используются компьютерные классы, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета и/или учебные аудитории, укомплектованные мебелью и техническими средствами обучения.

Самостоятельная работа. Помещения для самостоятельной работы оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и доступом к электронной информационно-образовательной среде университета. Для организации самостоятельной работы обучающихся используются:

- компьютерные классы университета;
- библиотека (медиазал), имеющая места для обучающихся, оснащенные компьютерами с доступом к базам данных и сети «Интернет».

Электронная информационно-образовательная среда университета (ЭИОС). Каждый обучающийся в течение всего периода обучения обеспечен индивидуальным неограниченным доступом к электронной информационно-образовательной среде университета (ЭИОС) http://sdo.tolgas.ru/из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети «Интернет», как на территории университета, так и вне ее.

6. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБУЧЕНИЯ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

При необходимости рабочая программа дисциплины может быть адаптирована для обеспечения образовательного процесса инвалидов и лиц с ограниченными возможностями здоровья, в том числе для дистанционного обучения. Для этого требуется заявление студента (его законного представителя) и заключение психолого-медико-педагогической комиссии (ПМПК).

В случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида, могут предлагаться следующие варианты восприятия учебной информации с учетом их индивидуальных психофизических особенностей, в том числе с применением электронного обучения и дистанционных технологий:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации.
- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.
- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.

7. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (ФОНД ОЦЕНОЧНЫХ СРЕДСТВ) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

7.1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта в ходе текущего контроля успеваемости

Типовые задания к практическим (семинарским) занятиям

5 семестр

Практическая работа №1. Построение диаграммы Вариантов использования и диаграммы. Последовательности и генерация кода.

Практическая работа №2. Построение диаграммы Кооперации и диаграммы Развертывания и генерация кода.

Практическая работа №3. Построение диаграммы Деятельности, диаграммы Состояний и диаграммы Классов и генерация кода.

Практическая работа №4. Построение диаграммы компонентов и генерация кода.

Практическая работа №5. Построение диаграмм потоков данных и генерация кода.

6 семестр

Практическая работа №1. Обоснование выбора технических средств.

Практическая работа №2. Стоимостная оценка проекта.

Практическая работа №3. Построение и обоснование модели проекта.

Практическая работа №4. Установка и настройка системы контроля версий с разграничением ролей.

Практическая работа №5. Проектирование и разработка интерфейса пользователя.

Практическая работа №6. Разработка графического интерфейса пользователя.

Практическая работа №7. Реализация алгоритмов обработки числовых данных. Отладка приложения.

Практическая работа №8. Реализация алгоритмов поиска. Отладка Приложения.

Практическая работа №9. Реализация обработки табличных данных. Отладка приложения.

Практическая работа №10. Разработка и отладка генератора случайных символов.

Практическая работа №11. Разработка приложений для моделирования процессов и явлений. Отладка приложения.

Практическая работа №12. Интеграция модуля в информационную систему.

Практическая работа №13. Программирование обмена сообщениями между модулями.

Практическая работа №14. Организация файлового ввода-вывода данных.

Практическая работа №15. Разработка модулей экспертной системы.

Практическая работа №16. Создание сетевого сервера и сетевого клиента.

Типовые вопросы (задания) для устного (письменного) опроса

5 семестр

1. Структура CASE-средства. Структура среды разработки. Основные возможности.
2. Основные инструменты среды для создания, исполнения и управления информационной системой. Выбор средств обработки информации.
3. Организация работы в команде разработчиков. Система контроля версий: совместимость, установка, настройка.
4. Обеспечение кроссплатформенности информационной системы.
5. Сервисно - ориентированные архитектуры.
6. Интегрированные среды разработки для создания независимых программ.
7. Особенности объектно-ориентированных и структурных языков программирования.
8. Разработка сценариев с помощью специализированных языков.

6 семестр

1. Обоснование и осуществление выбора модели построения или модификации информационной системы.
2. Обоснование и осуществление выбора средства построения информационной системы и программных средств.

3. Построение архитектуры проекта. Шаблон проекта.
4. Определение конфигурации информационной системы. Выбор технических средств.
5. Формирование репозитория проекта, определение уровня доступа в системе контроля версий. Распределение ролей.
6. Настройки среды разработки.
7. Мониторинг разработки проекта. Сохранение версий проекта.
8. Требования к интерфейсу пользователя. Принципы создания графического пользовательского интерфейса (GUI).
9. Понятие спецификации языка программирования. Синтаксис языка программирования. Стиль программирования.
10. Основные конструкции выбранного языка программирования. Описание переменных, организация ввода-вывода, реализация типовых алгоритмов.
11. Создание сетевого сервера и сетевого клиента.
12. Разработка графического интерфейса пользователя.
13. Отладка приложений. Организация обработки исключений.
14. Виды, цели и уровни интеграции программных модулей.
15. Выбор источников и приемников данных, сопоставление объектов данных.
16. Транспортные протоколы. Стандарты форматирования сообщений.
17. Организация файлового ввода-вывода.
18. Процесс отладки. Отладочные классы.
19. Спецификация настроек типовой ИС.

Типовые тестовые задания

1. Выберите правильные утверждения:
-: класс — это тип данных;
+: объект класса может использоваться всюду, где используется объект подкласса;
-: объект класса можно создать только один раз;
-: на объект класса может не ссылаться объектная переменная.

2. Вставьте на место прочерка название одного из принципов ООП так, чтобы получилось верное определение:_____ — это объединение данных и методов, предназначенных для манипулирования этими данными в новом типе — классе.
+: наследование
-: полиморфизм
-: позднее связывание
-: инкапсуляция

3. Дан код:

```
String s; // 1
if ((s = "java") == "java") {// 2
        System.out.println (s+ " true");
} else {
        System.out.println (s+ " false");
}
```

Что будет результатом компиляции и запуска этого кода?
+: ошибка компиляции в строке 1, переменная не проинициализирована
-: ошибка компиляции в строке 2, неправильное выражение для оператора if
-: на консоль выведется java true
-: на консоль выведется java false

4. По типу результата все методы можно разделить на:
-:статические и не статические
-:константы и переменные
+:с возвращаемым значением и без
-:целочисленные и дробные

-:с аргументами и без

5. Какой класс является суперклассом для всех классов:
-:Main
-:MainClass
+:Object
-:Super
-:Class

6. В какой кодировке хранятся символы в переменной типа char:
-:UTF-32
+:UTF-16
-:UTF-8
-:ASCII

7. Результат сравнения - это:
-:Объект
-:Число
-:Символ
+:Логическое значение
-:Строка

8. Что появится в консоли в результате работы программы:
<code>
byte a = 127;
a++;
System.out.println(a);
</code>
-:0
-:127
-:Ошибка времени исполнения
-:128
+:-128

9. Какой из способов объявления массива некорректен:
-:int a[][] = new int[5][5];
-:int[][] a = new int[5][5];
+: []int[] a = new int[5][5];
-:int[] a[] = new int[5][5];

10. Можно ли использовать оператор return в void методе:
+:Да
-:Нет

Типовые задания для контрольной работы

Для всех вариантов:

Задание 1. Разработать приложение на Java.

Задание 2. Проанализировать предметную область и выявить возможные функции для автоматизации их программной системой. Построить диаграмму вариантов использования для проектируемой системы.

Задание 3. В соответствии с заданием на проектирование провести анализ предметной области, выявить основные классы и их взаимосвязи. Описать структуру проекта с помощью диаграммы классов.

Задание 4. Построить диаграммы последовательности для всех вариантов использования проектируемой системы

Вариант 1. Разработка приложения для предметной области «Учёт горюче-смазочных материалов на автозаправочной станции» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о поступлении и реализации горюче-смазочных материалов на автозаправочных станциях г. Тольятти и Самарской области. Структура приложения обязательно должна включать следующие классы, созданные студентом: горюче-смазочный материал, автозаправочная станция и др.

Вариант 2. Разработка приложения для предметной области «Учёт товаров в магазине» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о поступлении и реализации товаров некоторого магазина. Структура приложения обязательно должна включать следующие классы, созданные студентом: товар, производитель, документ, поступление товара, реализация товара и др.

Вариант 3. Разработка приложения для предметной области «Организация учебного процесса в ВУЗе» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения об организации и диспетчеризации учебного процесса в ВУЗе. Структура приложения обязательно должна включать следующие классы, созданные студентом: академическая группа, специальность, дисциплина, аудитория, преподаватель и др.

Вариант 4. Разработка приложения для предметной области «Аквариум» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о динамике растительного и животного мира, обитающего в аквариуме. Структура приложения обязательно должна включать следующие классы, созданные студентом: рыбы, породы рыб, растения (водоросли), разновидности растений, условия обитания (относятся к растениям и рыбам), питание и др.

Вариант 5. Разработка приложения для предметной области «Учёт средств вычислительной техники на предприятии» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту средств вычислительной техники. Структура приложения обязательно должна включать следующие классы, созданные студентом: средство вычислительной техники, компонент средства вычислительной техники (ВТ), производитель средства ВТ, продавец средства ВТ, гарантийное обслуживание, пользователь средства ВТ, структурное подразделение и др.

Вариант 6**.** Разработка приложения для предметной области «Учёт основных средств предприятия» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту основных средств предприятия. Структура приложения обязательно должна включать следующие классы, созданные студентом: карточка основного средства, группа основного средства, амортизация основного средства, списание основного средства, реновация основного средства, структурное подразделение и др.

Вариант 7. Разработка приложения для предметной области «Учёт транспортных средств и их владельцев» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту транспортных средств и их владельцев. Структура приложения обязательно должна включать следующие классы, созданные студентом: транспортное средство (ТС), категория транспортного средства, техосмотр, владелец ТС (юридическое или физическое лицо), топливо, модель ТС и др.

Вариант 8. Разработка приложения для предметной области «Учёт рек с точки зрения народнохозяйственного значения» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту рек, протекающих на территории некоторого региона. Структура приложения обязательно должна включать следующие классы, созданные студентом: реки, экологическая характеристика, народнохозяйственное значение, регионы, предприятия, другие водоёмы и др.

Вариант 9. Разработка приложения для предметной области «Учёт растений в ботаническом саду» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту растений, произрастающих на территории некоторого ботанического сада. Структура приложения обязательно должна включать следующие классы, созданные студентом: растение, семейство, вид, род, назначение, территория произрастания, карточка учёта растений в ботаническом саду, совместимость с другими видами и др.

Вариант 10. Разработка приложения для предметной области «Учёт дикий животных в охотоведческом хозяйстве» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту диких животных, обитающих на территории охотничьего хозяйства. Структура приложения обязательно должна включать следующие классы, созданные студентом: дикое животное, вид диких животных, класс животных, корм, лицензии на отстрел, карточка визуального учёта животных и др.

Вариант 11. Разработка приложения для предметной области «Учёт домашних животных в фермерском хозяйстве» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту домашних животных, содержащихся в фермерском хозяйстве. Структура приложения обязательно должна включать следующие классы, созданные студентом: домашнее животное, вид домашнего животного, порода, класс животных, корм, карточка учёта животных, владелец и др.

Вариант 12. Разработка приложения для предметной области «Сведения о памятниках истории и архитектуры» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о памятниках истории и архитектуры, представляющих историческую и государственную ценность. Структура приложения обязательно должна включать следующие классы, созданные студентом: памятник, регион, создатель памятника, вид памятника, экспертная оценка состояния и др.

Вариант 13. Разработка приложения для предметной области «Сбор сведений о писателях и их литературных произведениях» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о писателях и их литературных произведениях. Структура приложения обязательно должна включать следующие классы, созданные студентом: писатель (автор), произведение, жанр, страна (где проживал автор), язык (на котором написано или было переведено произведение), издание произведения (тираж), издательство и др.

Вариант 14. Разработка приложения для предметной области «Сведения о промышленных предприятиях» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о промышленных предприятиях. Структура приложения обязательно должна включать следующие классы, созданные студентом: промышленное предприятие, форма собственности, сведения о владельце, продукция, партнёры, конкуренты, отрасль экономики (к которой относится) и др.

Вариант 15. Разработка приложения для предметной области «Учёт абитуриентов, поступающих в ВУЗ» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения об абитуриентах ВУЗа. Структура приложения обязательно должна включать следующие классы, созданные студентом: абитуриент, дисциплина, экзамен, специальность, экзаменационный лист, документ и др.

Вариант 16. Разработка приложения для предметной области «Математика» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и систематизировать сведения по разделам математики. Структура приложения обязательно должна включать следующие классы, созданные студентом: раздел математики (дисциплина), аксиомы, теоремы, леммы, сведения об учёном-математике, задача, формула и др.

Вариант 17. Разработка приложения для предметной области «Учёт вкладов, помещённых в банк» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о деятельности отдела банка, работающего с вкладчиками. Структура приложения обязательно должна включать следующие классы, созданные студентом: вкладчик, вклад, тип вклада, сведения о банке, валюта, операции с вкладом и др.

Вариант 18. Разработка приложения для предметной области «Учёт пациентов в регистратуре поликлиники» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту пациентов в регистратуре городской поликлинике. Структура приложения обязательно должна включать следующие классы, созданные студентом: карточка больного (титульный лист), записи карточки, врач, журнал регистрации врача на дом, лекарство, заболевание и др.

Вариант 19. Разработка приложения для предметной области «Учёт объектов библиотечного фонда» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту книг в библиотеке. Структура приложения обязательно должна включать следующие классы, созданные студентом: карточка читателя (титульный лист), записи карточки, объект библиотечного фонда (книга, газета, журнал, ...), издательство, карточка каталога, вид объекта библиотечного фонда, библиотечный раздел и др.

Вариант 20. Разработка приложения для предметной области «Учёт заказов клиентов в ресторане» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения по учёту заказов от клиентов в ресторане. Структура приложения обязательно должна включать следующие классы, созданные студентом: блюдо, группа блюд, заказ, клиент, столик, меню заказа (ценник) и др.

Вариант 21. Разработка приложения для предметной области «Учёт клиентов и мобильных телефонов компании, предоставляющей услуги мобильной связи» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о клиентах и мобильных телефонах некоторой компании, предоставляющей услуги мобильной связи. Структура приложения обязательно должна включать следующие классы, созданные студентом: мобильный телефон, модель, владелец мобильного телефона, услуга, тариф, платёж и др.

Вариант 22. Разработка приложения для предметной области «Деятельность страховой компании» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о финансовой деятельности компании. Учитывается, что страховая компания имеет множество филиалов по всей стране. При заключении договора на страхование необходимо зафиксировать дату заключения договора, страховую сумму, вид страхования, филиал в котором заключался данный договор, тарифную ставку. Также обратить внимание на то, что договоры заключают непосредственно страховые агенты, соответственно необходимо хранить всю информацию о агентах и филиале, в котором они работают.
Структура приложения обязательно должна включать следующие классы, созданные студентом: Договоры (Номер, дата заключения, страховая сумма, тарифная ставка, филиал, вид страхования), Вид страхования (Наименование), Филиал (Наименование, адрес, телефон) и др.

Вариант 23. Разработка приложения для предметной области «Деятельность ломбарда» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее собирать и накапливать сведения о финансовой деятельности компании. Задачей является отслеживание финансовой деятельности компании. К специалисту компании клиенты с целью получения денежных средств под залог определенных товаров. При обращении клиента специалист производит оценку стоимости и качества принесенного вам товара, определяет сумму, которую компания готова выдать клиенту и определяет свои комиссионные, также определяете срок возврата денежных средств. Если клиент согласен, то специалист спрашиваете данные клиента (ФИО, паспортные данные) и все договоренности фиксируются документально. В случае, если клиент в указанный срок не производит возврат денег, товар переходит в собственность ломбарда и может быть выставлен на продажу по цене меньшей или большей, чем была заявлена при сдаче.
Структура приложения обязательно должна включать следующие классы, созданные студентом: клиент (ФИО, номер, серия паспорта, дата выдачи паспорта, кем выдан паспорт), категория товаров (Название, примечание), договор о сдаче в ломбард (Клиент, описание товара, дата сдачи, категория товара, дата возврата, сумма выданные средств, комиссионные) и др.

Вариант 23. Разработка приложения для предметной области «Деятельность банка по выдаче кредита» с применением языка программирования Java.
Задание 1. Разработать приложение на Java, позволяющее автоматизировать деяьельность руководителя информационно-аналитического отдела коммерческого банка. Обязанностью данного специалиста является отслеживание динамики кредитного отдела для юридических лиц. Кредит может получить юридическое лицо при предоставлении таких сведений: название, вид собственности, адрес, телефон, контактное лицо. Каждый кредит делится на несколько основных видов, в зависимости от процентной ставки, срока возврата и условий получения кредита, каждый из данных видов имеет свое название. Каждый факт выдачи кредита банком регистрируется, при этом фиксируется сумма кредита, клиент и дата выдачи.
Структура приложения обязательно должна включать следующие классы, созданные студентом: виды кредитов (Название, условия получения, ставка, срок), клиент (Название, вид собственности, адрес, телефон, контактное лицо), кредит (Вид кредита, клиент, сумма, дата выдачи).

7.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта в ходе промежуточной аттестации

Форма проведения промежуточной аттестации по МДК: *дифференцированный зачет / контрольная работа (по результатам накопительного рейтинга или в форме компьютерного тестирования).*

Устно-письменная форма по экзаменационным билетам предполагается, как правило, для сдачи академической задолженности.

Защита контрольной работы. *Результаты защиты контрольной работы*

выставляются по пятибалльной системе оценивания ("отлично", "хорошо", "удовлетворительно", "неудовлетворительно") с обязательным проставлением количества баллов, набранных в соответствии с балльно-рейтинговой системой (по стобалльной шкале).

Перечень вопросов и заданий для подготовки к дифференцированному зачету (ОК 01, ОК 02, ОК 03, ОК 04, ОК 09, ОК 10, ПК 5.1, ПК 5.2, ПК 5.3, ПК 5.4, ПК 5.5, ПК 5.6, ПК 5.7)**:**

Контрольные вопросы и задания

1. Структура CASE-средства.
2. Структура среды разработки. Основные возможности.
3. Основные инструменты среды для создания, исполнения и управления информационной системой.
4. Выбор средств обработки информации.
5. Организация работы в команде разработчиков.
6. Система контроля версий:
совместимость, установка, настройка.
7. Обеспечение кроссплатформенности информационной системы.
8. Сервисно - ориентированные архитектуры.
9. Интегрированные среды разработки для создания независимых программ.
10. Особенности объектно-ориентированных и структурных языков программирования.
11. Разработка сценариев с помощью специализированных языков.
12. Обоснование и осуществление выбора модели построения или модификации информационной системы.
13. Обоснование и осуществление выбора средства построения информационной системы и программных средств.
14. Построение архитектуры проекта. Шаблон проекта.
15. Определение конфигурации информационной системы. Выбор технических средств.
16. Формирование репозитория проекта, определение уровня доступа в системе контроля версий. Распределение ролей.
17. Настройки среды разработки.
18. Мониторинг разработки проекта. Сохранение версий проекта.
19. Требования к интерфейсу пользователя. Принципы создания графического пользовательского интерфейса (GUI).
20. Понятие спецификации языка программирования.
21. Синтаксис языка программирования.
22. Стиль программирования.
23. Основные конструкции выбранного языка программирования.
24. Описание переменных, организация ввода-вывода, реализация типовых алгоритмов.
25. Создание сетевого сервера и сетевого клиента.
26. Разработка графического интерфейса пользователя.
27. Отладка приложений.
28. Организация обработки исключений.
29. Виды, цели и уровни интеграции программных модулей.
30. Выбор источников и приемников данных, сопоставление объектов данных.
31. Транспортные протоколы. Стандарты форматирования сообщений.
32. Организация файлового ввода-вывода.
33. Процесс отладки. Отладочные классы.
34. Спецификация настроек типовой ИС.

Примерный тест для итогового тестирования (ОК 01, ОК 02, ОК 03, ОК 04, ОК 09, ОК 10, ПК 5.1, ПК 5.2, ПК 5.3, ПК 5.4, ПК 5.5, ПК 5.6, ПК 5.7)**:**

1. Выберите правильные утверждения:
-: класс — это тип данных;

+: объект класса может использоваться всюду, где используется объект подкласса;
-: объект класса можно создать только один раз;
-: на объект класса может не ссылаться объектная переменная.

2. Вставьте на место прочерка название одного из принципов ООП так, чтобы получилось верное определение:_____ — это объединение данных и методов, предназначенных для манипулирования этими данными в новом типе — классе.
+: наследование
-: полиморфизм
-: позднее связывание
-: инкапсуляция

3. Дан код:

```
String s; // 1
if ((s = "java") == "java") {// 2
        System.out.println (s+ " true");
} else {
        System.out.println (s+ " false");
}
```

Что будет результатом компиляции и запуска этого кода?
+: ошибка компиляции в строке 1, переменная не проинициализирована
-: ошибка компиляции в строке 2, неправильное выражение для оператора if
-: на консоль выведется java true
-: на консоль выведется java false

4. По типу результата все методы можно разделить на:
-:статические и не статические
-:константы и переменные
+:с возвращаемым значением и без
-:целочисленные и дробные
-:с аргументами и без

5. Какой класс является суперклассом для всех классов:
-:Main
-:MainClass
+:Object
-:Super
-:Class

6. В какой кодировке хранятся символы в переменной типа char:
-:UTF-32
+:UTF-16
-:UTF-8
-:ASCII

7. Результат сравнения - это:
-:Объект
-:Число
-:Символ
+:Логическое значение
-:Строка

8. Что появится в консоли в результате работы программы:
<code>
byte a = 127;

a++;
System.out.println(a);
</code>
-:0
-:127
-:Ошибка времени исполнения
-:128
+:-128

9. Какой из способов объявления массива некорректен:
-:int a[][] = new int[5][5];
-:int[][] a = new int[5][5];
+: []int[] a = new int[5][5];
-:int[] a[] = new int[5][5];

10. Можно ли использовать оператор return в void методе:
+:Да
-:Нет

11. Являются ли два метода перегруженными, если у них одинаковые имена, разный набор аргументов и разный тип возвращаемого значения:
-:Нет
+:Да

12. Укажите правильную форму метода main:
+:public static void main(String[] args)
-:public void main(String[] args)
-:public static void main()
-:public static void Main(String[] args)
-:public int main(String[] args)

13. Какое значение будет сохранено в строке s?
<code>
String s = 2 + 2 + " = value";
</code>
-:ошибка компиляции
+:"4 = value"
-:"22 = value"
-:"2 + 2 = value"
-:"2 2 = value"

14. Как узнать количество элементов в массиве?
-:arrayName.size()
-:size(arrayName)
+:arrayName.length()
-:arrayName.length
-:length(arrayName)

15. Какой цикл имеет инициализатор, условие и шаг?
+:for
-:infinite loop
-:do-while
-:for-earch
-:while

16. Можно ли переопределять (override) private методы?
+:Нет
-:Да

17. Использование геттеров и сеттеров относится к принципу?
+:Инкапсуляция
-:Наследование
-:Полиморфизм

19. Инициализированная переменная в Java имеет:
-:Имя, первоначальное значение
-:Тип, имя
-:Имя
-:Тип, первоначальное значение
+:Тип, имя, первоначальное значение

20. Для объявления переменной в Java достаточно задать:
-:Тип, имя, первоначальное значение
+:Тип, имя
-:Тип, первоначальное значение
-:Имя
-:Имя, первоначальное значение

21. Для массивов в Java верно:
-:Изменяемый состав элементов разных типов
-:Фиксированное количество элементов разных типов
+:Фиксированное количество элементов одного типа
-:Изменяемый состав элементов одного типа

22. Может ли значение статического поля у объектов одного классы быть разным:
-:Да
+:Нет

23. Возможно ли унаследовать класс от двух суперклассов (множественное наследование):
-:Да
+:Нет

24. Какие категории типов данных поддерживаются в Java:
-:Переменные, примитивные
-:Ссылочные (объектные)
-:Примитивные
+:Примитивные, ссылочные (объектные)
-:Переменные, постоянные

25. Вызвать не статический метод или свойство из другого класса можно:
+:Только создав экземпляр класса, в котором они реализованы
-:Не создавая экземпляр класса, в котором они реализованы
-:Только из зависимых классов
-:Из любого класса в том же пакете
-:Из любого метода в этом же классе

26. К какому принципу ООП относятся понятия прав доступа и модификаторов public, protected, package-private и private?

-:Полиморфизм
-:Реализация
-:Абстракция
-:Наследование
+:Инкапсуляция

27. Можно ли в одном классе объявить больше одного метода main()?
-:Нет
+:Да

28. Возможно ли получить доступ к отдельным символам или диапазонам символов строки?
-:Да, символьные индексы начинаются с 'A'
-:Да, числовые индексы начинаются с 1
-:Да, символьные индексы начинаются с 'a'
-:Нет, доступ к отдельным символам строки получить нельзя
+:Да, числовые индексы начинаются с 0

29. Как Java CodeConvention рекомендует называть классы и методы?
-:Классы пишутся заглавными буквами, методы целиком строчными
-:Классы пишутся строчными буквами, методы целиком заглавными
-:Классы начинаются со строчной буквы, методы с заглавной
-:Классы начинаются с заглавной буквы, методы тоже с заглавной
+:Классы начинаются с заглавной буквы, методы со строчной

30. Если в сигнатуре метода указано void, то ключевое слово return в теле метода:
-:Не обязательно, может быть указано с любым значением
+:Не обязательно, но может быть указано в качестве прерывания работы метода
-:Обязательно, но без какого-либо значения
-:Обязательно как минимум один раз с любым значением
-:Обязательно как минимум один раз без какого-либо значения

Регламент проведения промежуточной аттестации в форме компьютерного тестирования

Кол-во заданий в банке вопросов	Кол-во заданий, предъявляемых студенту	Время на тестирование, мин.
не менее 60 или указывается конкретное количество тестовых заданий	*30*	*30*

Полный фонд оценочных средств для проведения промежуточной аттестации в форме компьютерного тестирования размещен в банке вопросов данного курса дисциплины в ЭИОС университета http://sdo.tolgas.ru/.

В ходе подготовки к промежуточной аттестации обучающимся предоставляется возможность пройти тест самопроверки. Тест для самопроверки по дисциплине размещен в ЭИОС университета http://sdo.tolgas.ru/ в свободном для студентов доступе.

АННОТАЦИЯ

МДК.05.02 «Разработка кода информационных систем»

Междисциплинарный курс «Разработка кода информационных систем» относится к профессиональному циклу основной профессиональной образовательной программы.

Целью освоения междисциплинарного курса является формирование у обучающихся следующих компетенций:

Код компетенции	Наименование компетенции
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.
ОК 02	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
ОК 03	Планировать и реализовывать собственное профессиональное и личностное развитие.
ОК 04	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
ОК 09	Использовать информационные технологии в профессиональной деятельности.
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языке.
ПК 5.1	Собирать исходные данные для разработки проектной документации на информационную систему.
ПК 5.2	Разрабатывать проектную документацию на разработку информационной системы в соответствии с требованиями заказчика.
ПК 5.3	Разрабатывать подсистемы безопасности информационной системы в соответствии с техническим заданием.
ПК 5.4	Производить разработку модулей информационной системы в соответствии с техническим заданием.
ПК 5.5	Осуществлять тестирование информационной системы на этапе опытной эксплуатации с фиксацией выявленных ошибок кодирования в разрабатываемых модулях информационной системы.
ПК 5.6	Разрабатывать техническую документацию на эксплуатацию информационной системы.
ПК 5.7	Производить оценку информационной системы для выявления возможности ее модернизации.

В результате освоения междисциплинарного курса обучающийся должен:

иметь практический опыт в:

управлении процессом разработки приложений с использованием инструментальных средств;

обеспечении сбора данных для анализа использования и функционирования информационной системы;

программировании в соответствии с требованиями технического задания;

использовании критериев оценки качества и надежности функционирования информационной системы;

применении методики тестирования разрабатываемых приложений;

определении состава оборудования и программных средств разработки информационной системы;

разработке документации по эксплуатации информационной системы;

проведении оценки качества и экономической эффективности информационной системы в рамках своей компетенции;

модификации отдельных модулей информационной системы.

уметь:

осуществлять постановку задач по обработке информации;

проводить анализ предметной области;

осуществлять выбор модели и средства построения информационной системы и программных средств;

использовать алгоритмы обработки информации для различных приложений;

решать прикладные вопросы программирования и языка сценариев для создания программ;

разрабатывать графический интерфейс приложения;

создавать и управлять проектом по разработке приложения;

проектировать и разрабатывать систему по заданным требованиям и спецификациям.

знать:

основные виды и процедуры обработки информации, модели и методы решения задач обработки информации;

основные платформы для создания, исполнения и управления информационной системой;

основные процессы управления проектом разработки;

основные модели построения информационных систем, их структуру, особенности и области применения;

методы и средства проектирования, разработки и тестирования информационных систем;

систему стандартизации, сертификации и систему обеспечения качества продукции.